Требования к верстке

Базовый шаблон для верстки: ​<https://d.pr/f/QE8QWo>

1.В папке с версткой страница index.html должна быть каталогом всех сверстанных

страниц.

2.Страницы в верстке должны быть перелинкованы между собой (все меню, ссылки на

синглы и т.д. должны вести на соответствующие страницы, чтобы по верстке можно

было “походить” и понять логику сайта.

3.Верстка должна быть responsive, кроссбраузерной и кроссплатформенной (актуальные

версии браузеров + одна версия назад)

a.баннера и любые div для вставки рекламного кода (пикселя) не должны быть

адаптивными (если в ТЗ не указана сетка адаптивности для DFP)

b.для тачскринов ховеры отключаем, чтобы тап сразу срабатывал, как клик

4.Верстка должна быть проверена на мобильном устройстве + указываем мета-атрибут

viewport (​должна быть задана минимальная и максимальная ширина​)

5.Обязательно наличие сброса стилей (reset). Можно использовать альтернативу в виде

normalize.css

6.Никаких CSS фреймворков типа bootstrap, foundation (только при наличии веских

аргументов)

7.Использовать SASS (.scss) + Gulp

8.Не злоупотреблять вложенностями SASS, дабы уменьшить размер CSS-файла на

выходе

9.Использовать BEM для нейминга CSS классов

10.При написании кода соблюдать отступы для наглядности в иерархии; снабжать

комментариями открытие и закрытие кода основных блоков, с указанием класса или ID

этого блока (<!-- .container -->, <!-- #header -->)

11.Именуем классы без нижнего подчеркивания (\_), используем дефис (-). Нижнее

подчеркивание используем для js скриптов.

12.Классы и ID следует называть на правильном английском: header, mail, testimonials, …

недопустимо называть транслитом: zagolovok, pochta, otzyvy, …

13.Все, что можно сделать без Javascript, — делать без него

14.Ориентируемся, что верстка будет использована для написания темы для WP, поэтому

сразу используем общепринятые названия стилей: ​<http://d.pr/JDl5>

15.Структурные блоки должны быть вынесены в отдельные scss файлы:

**\_header.scss \_content.scss \_typography.scss \_sidebar.scss \_footer.scss**

**\_variables.scss**

(переменные цветов, шрифтов, спрайта и других повторяющихся элементов и т.д.)

16.Файловая структура проекта:

**/assets**

**/scss**

**/css**

**/img**

**/del​** (картинки которые будут удалены при программировании)

 ​**/js**

**/min​** (минимизированные версии скриптов)

17.Использование в полном объеме тегов html5

​<https://www.w3schools.com/html/html5_semantic_elements.asp>

18.Фотографии сохраняются только в jpg, остальные картинки — png, иконки и все

остальные векторные элементы из макетов — svg!

19.jQuery и скрипты должны быть в подвале (библиотеку jQuery подключаем локально)

20.Для слайдеров использовать ​**slick.js​**. Слайдеры должны быть чувствительны к жестам

на тачскринах + responsive

21.Принцип адаптивности таблиц:

<https://css-tricks.com/examples/ResponsiveTables/responsive.php>

22.Селекты пишем вручную (обязательно с <input type=”hidden”>) или vue.js select

23.Если используем подключаемые шрифты, то Google Font. Если нужного нам шрифта

там нет, то подключаем через CSS import (Онлайн-конвертер шрифтов

[http://onlinefontconverter.com/​](http://onlinefontconverter.com/))

​**@import ‘fonts’ ​**— в файле​ **style.scss**

24.Все иконки — в svg для их преобразования в шрифт (иконки элементов должны

находиться в псевдо-элементе :before)

25.Если макет не подразумевает :hover или :active состояний (такое случается), то нужно

его оживлять самостоятельно: использовать transition + opacity + transform, ну или же

внутренний box-shadow (если речь о кнопках в состоянии :active). При этом абсолютно

каждый элемент, который является ссылкой, обязан менять свое состояние как

минимум при наведении. Читаем и вдохновляемся на ​<http://tympanus.net/codrops/>

26.Все ссылки, которые выполняют динамические действия на странице (т.е. если никаких

переходов на другие страницы не происходит), должны быть кнопками — ​**<button>**

27.SEO — ​<http://www.seochecklist.ru/>

a.на странице не более одного h1

b.расставлять правильно noindex, nofollow

c.не более двух h2

d.блок с заголовком должен иметь класс main-header

e.блок с контентом поста должен быть с классом main-content

f.Обязательно использовать стандарты Schema.org для элементов, которые

поддержаиваются (хлебные крошки, рейтинги, комментарии, товары и т.д.)

28.Должна быть страница со всем тестовым контентом и форматированием из

[http://www.wpfill.me/​](http://www.wpfill.me/) (кнопка Gimme everything) — это на этапе сдачи готового сайта

29.Придерживаться стандартов WP - ​<http://goo.gl/WTuIkr>

**30.Передача верстки на тестирование обязательно сопровождается скриншотами из**

**основных браузеров, где макет наложен на скрин верстки для визуального**

**сравнения pixel-perfect (пример: ​**[**http://d.pr/i/1alNU​**](http://d.pr/i/1alNU)**)**